

EXHIBIT A

Use of Declaring Code and Design of Java APIs in Android – A Technical Perspective

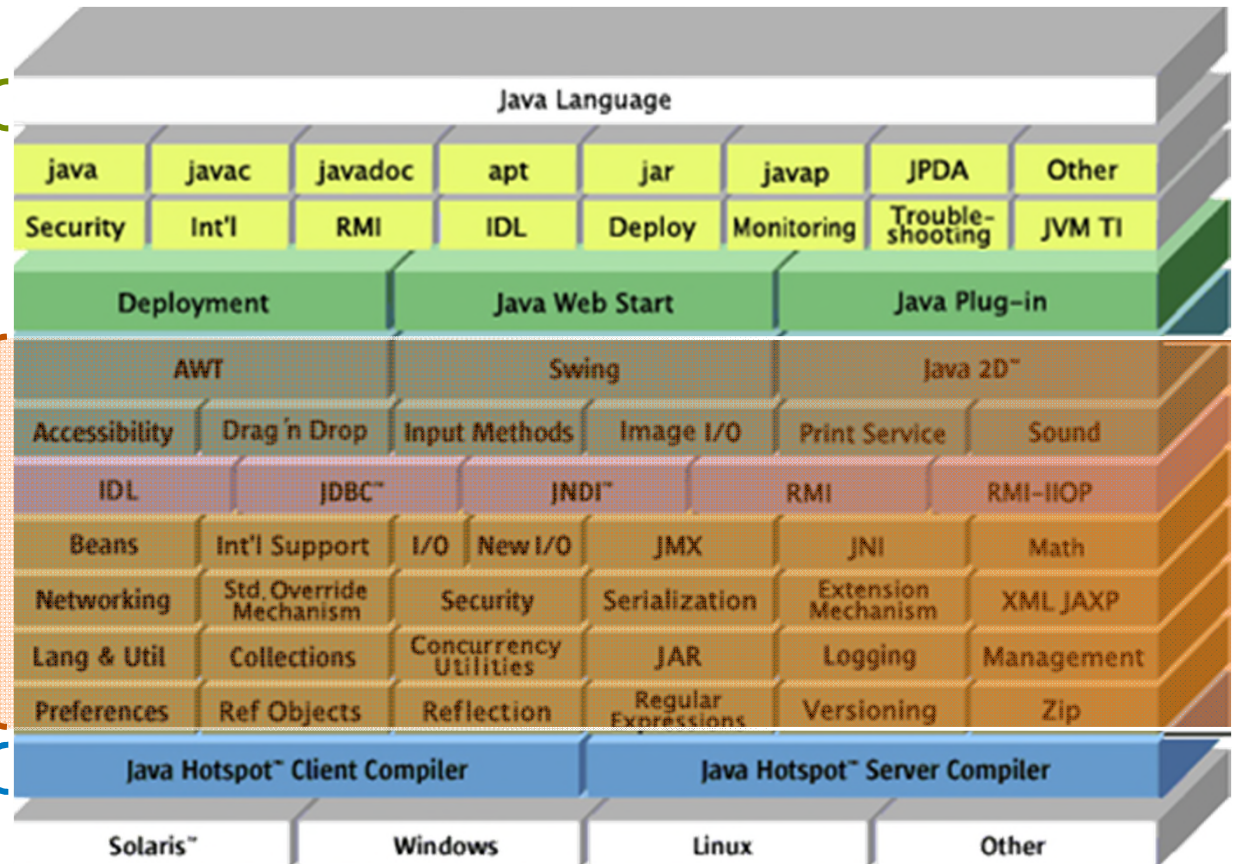
PROF. DOUGLAS C. SCHMIDT

MAY 17, 2016

Background: What Is the Java Platform?

Three Main Components – Each With Its Own Published Specification

Java Language
Popular object-oriented language {



Java API Packages
Pre-written Programs Providing Useful Functionality To Easily Create Feature-rich And Powerful Applications {

Java Virtual Machine
Runs Java apps on a wide range of operating systems {

The Java programming language, API Packages, and Virtual Machine are defined in three separate specifications

GOOGLE COPIED A KEY PART OF THE JAVA PLATFORM

Summary of Opinions

BASIS OF OPINION

The **declaring code and organization** of the 37 API packages **constitute the heart of the copyrighted work**

- The declaring code and design of the copied 37 Java APIs are critically important to developers

Google **did not transform the declaring code or organization** of the 37 API packages

- The purpose of the declaring code and design of the 37 Java APIs is the same
- Java-based mobile platforms existed before Android

The **declaring code and organization** of the 37 API packages **are creative and expressive**

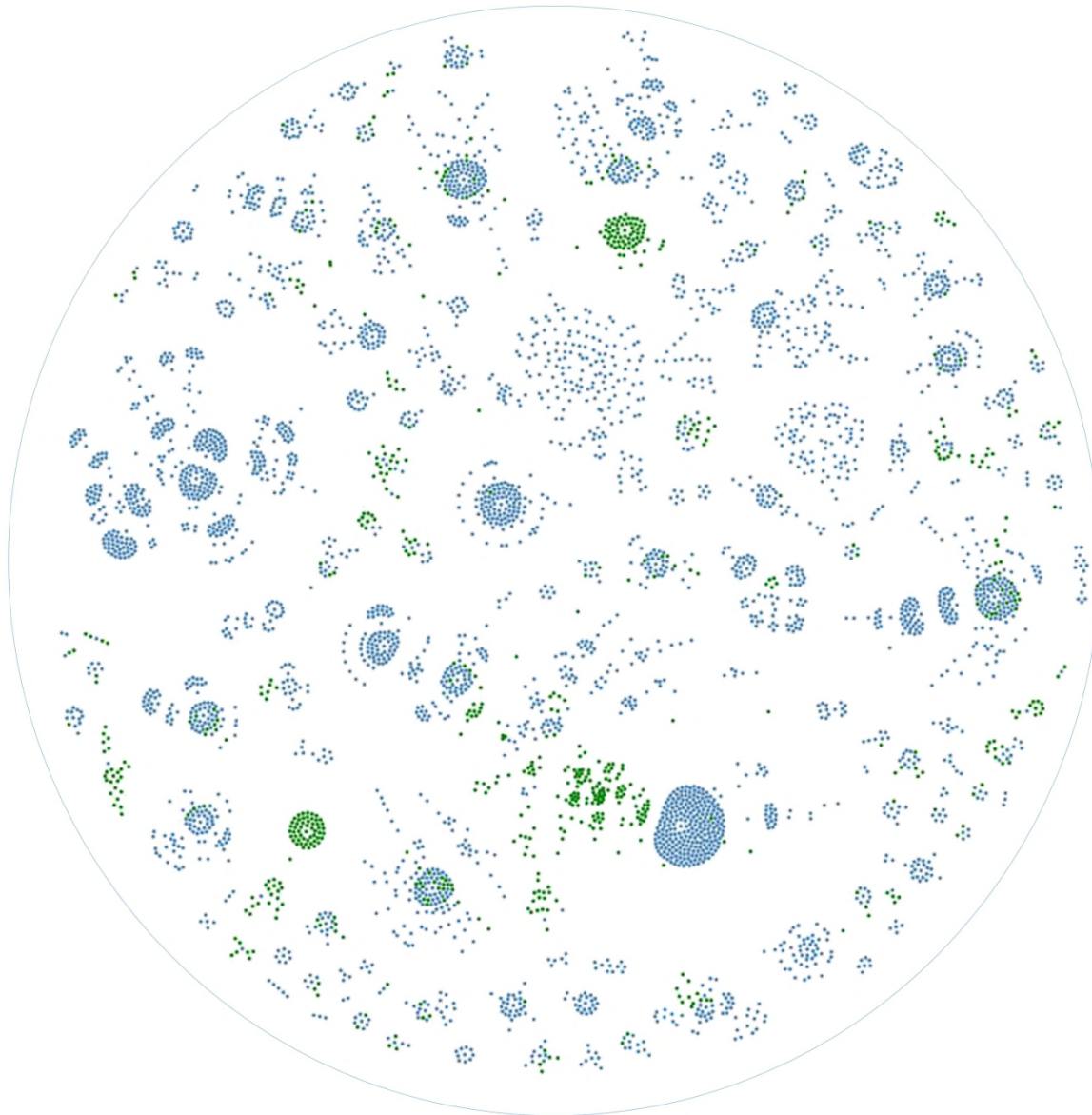
- The copied declaring code and design reflect creative choices and could have been achieved very differently

Android is **not interoperable or compatible** with Java

- Java apps won't run on Android; Android fails the Java compatibility test

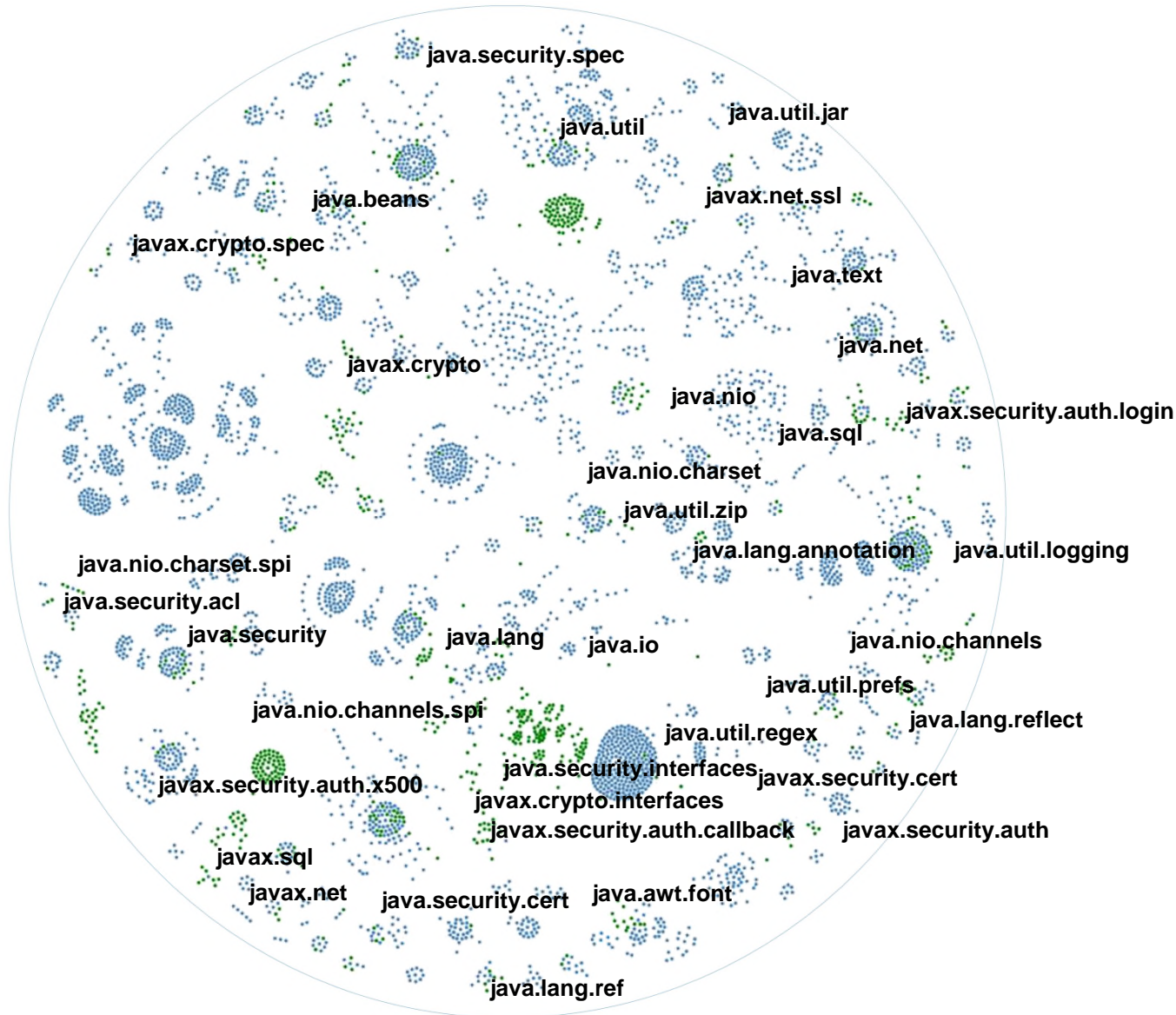
Software Map of the Java SE5 APIs

Classes and Interfaces



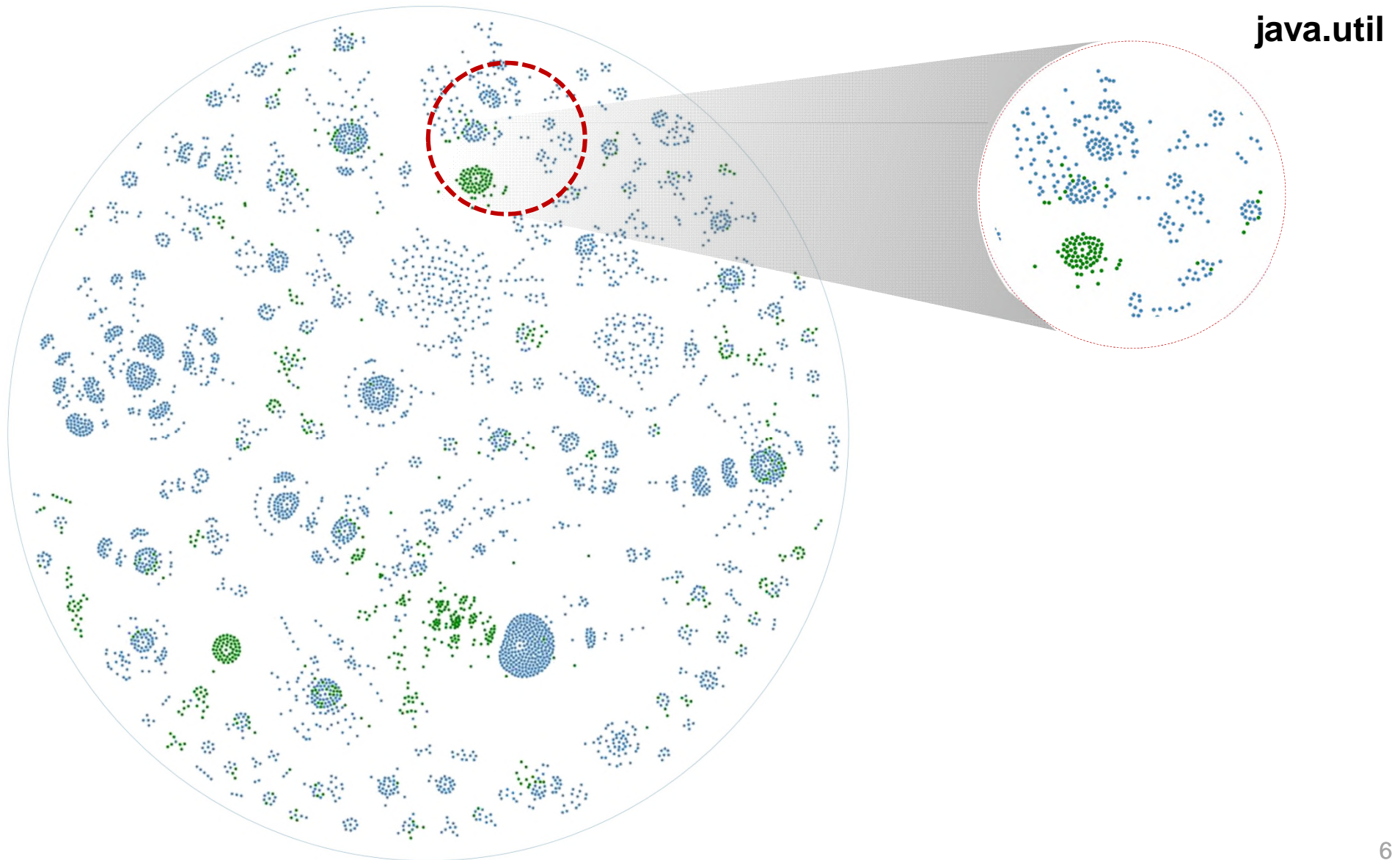
Software Map of the Java SE5 APIs

Packages, Classes, and Interfaces



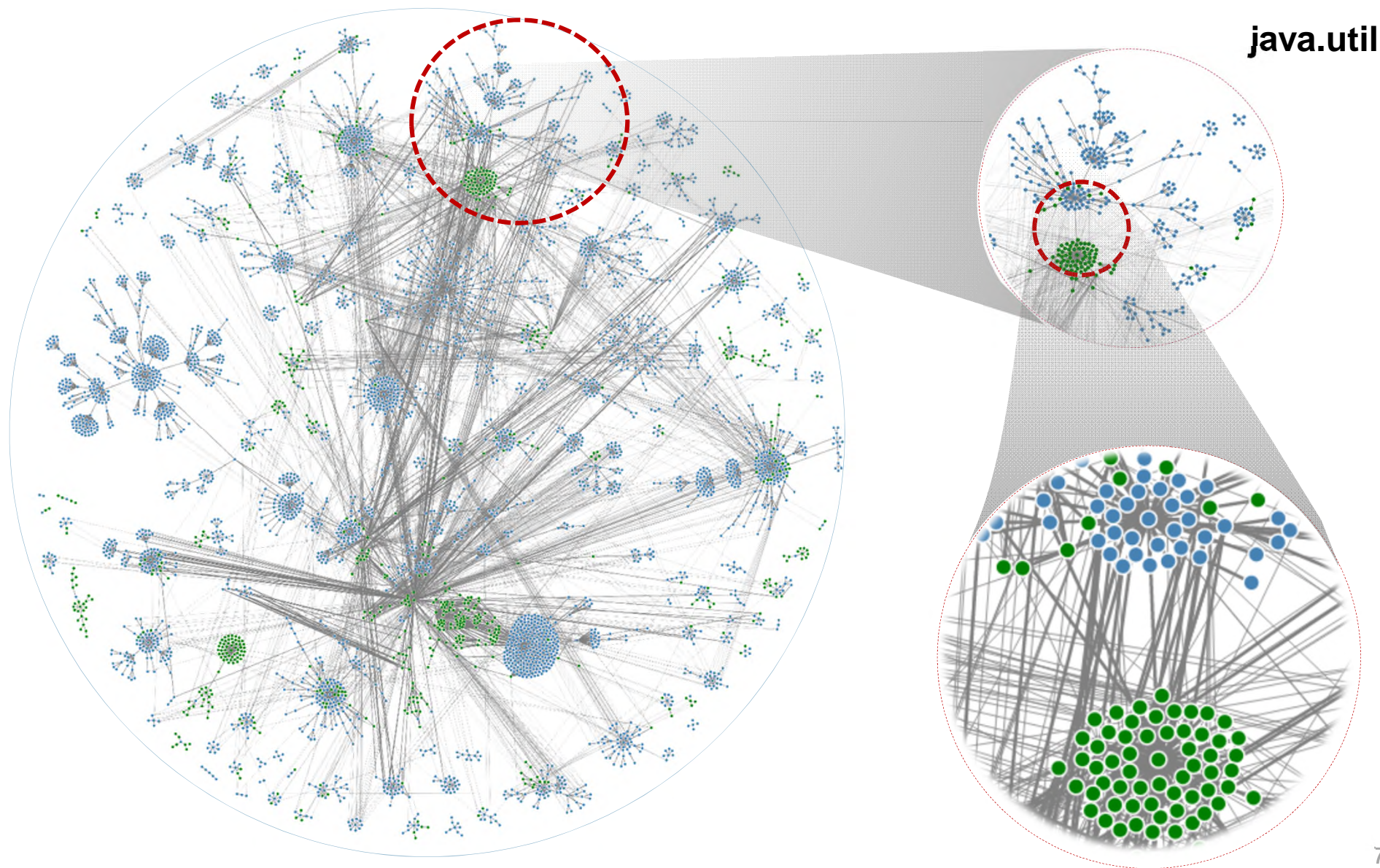
Software Map of the Java SE5 APIs

Packages, Classes and Interfaces



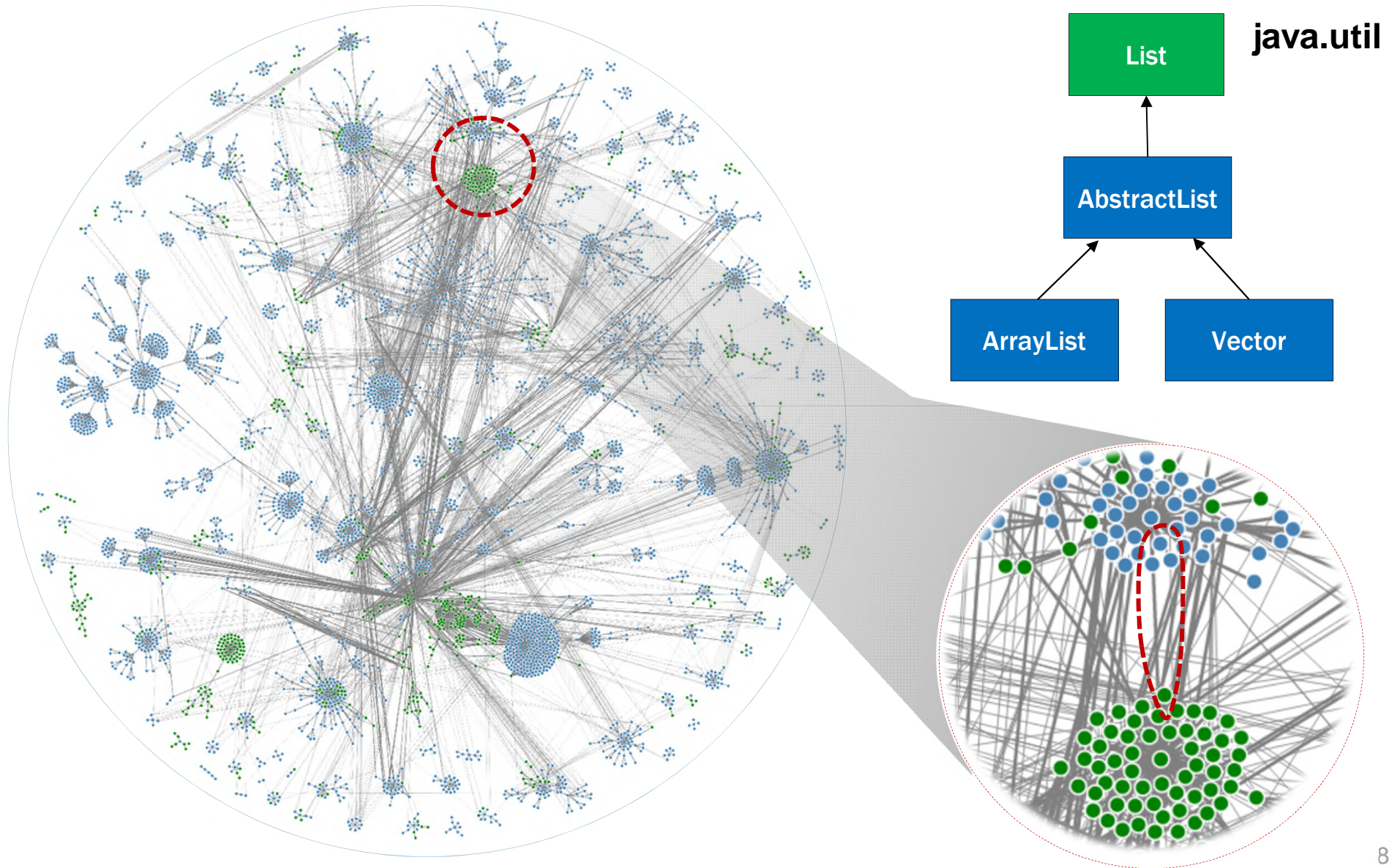
Software Map of the Java SE5 APIs

Class/Interface Relationships



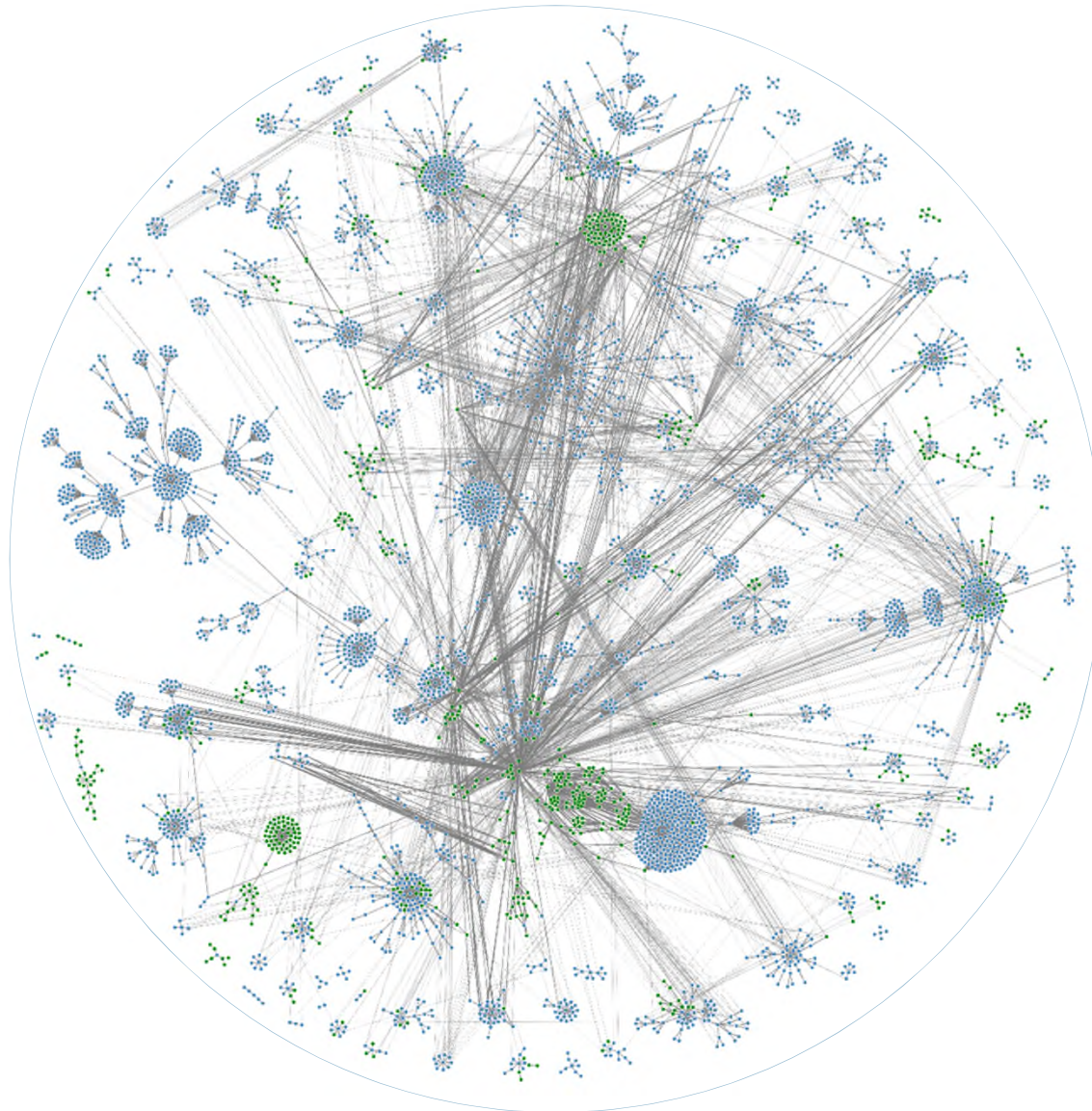
Software Map of the Java SE5 APIs

Class/Interface Relationships – java.util

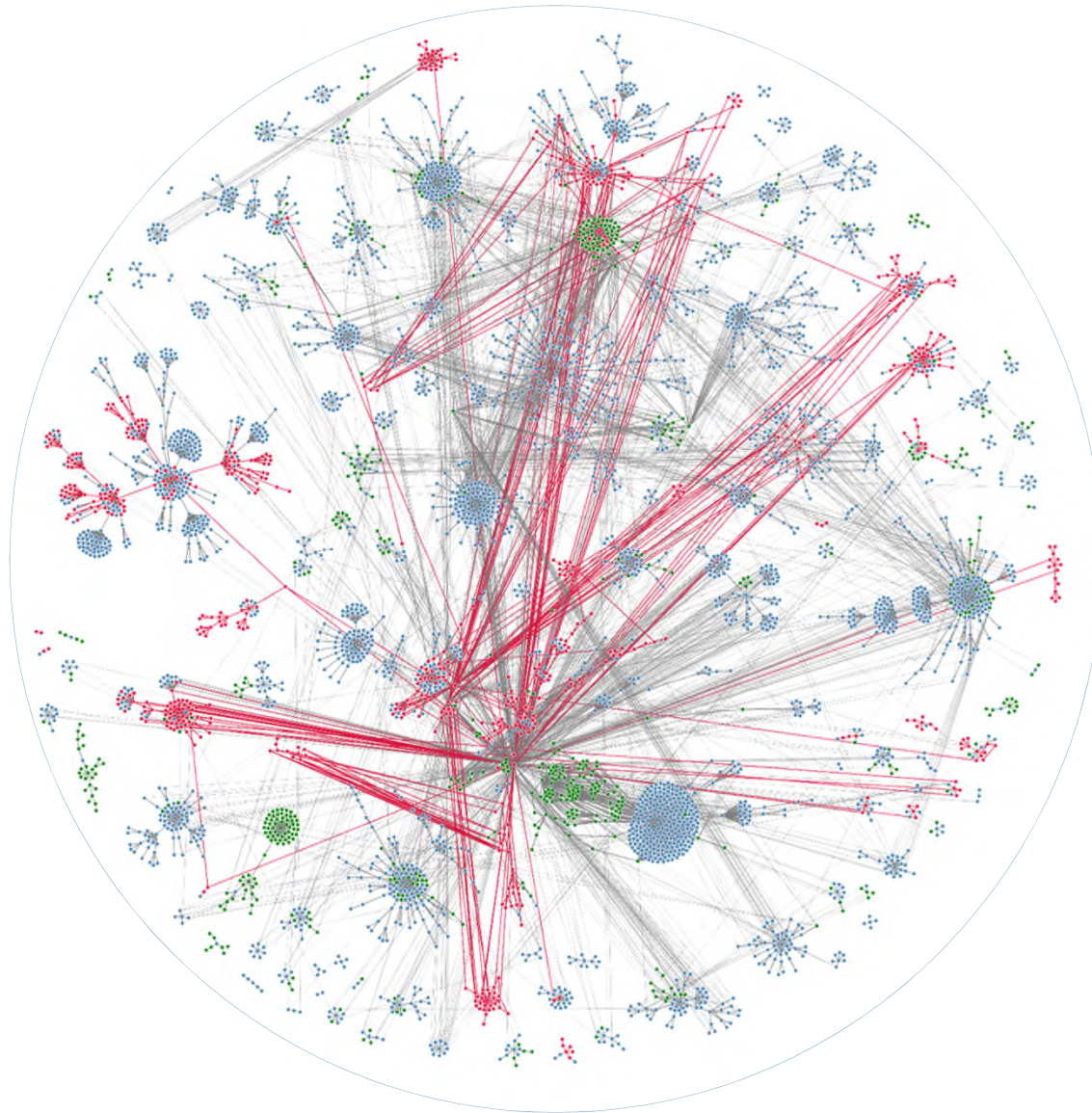


Software Map of the Java SE5 APIs

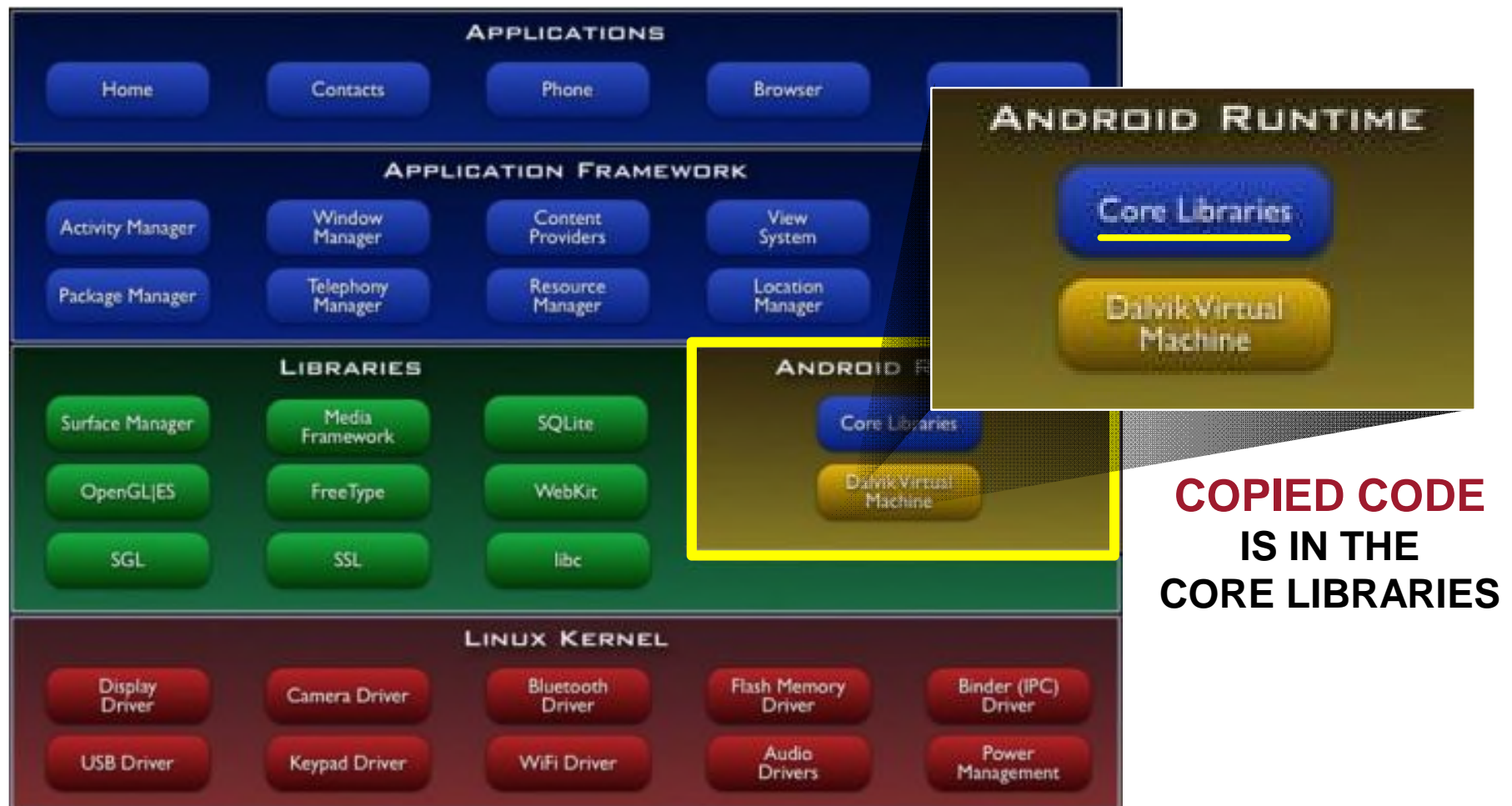
Relationships Between Classes and Interfaces



The Heart of the Java SE5 API is Found in Android



The Copied Code Is a Critical Part of Android



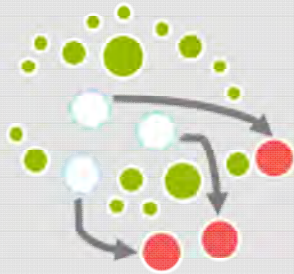
The Copied Code Is a Critical Part of Android

Platform dependency test highlights the dependence of the Android platform on the copied declaring code

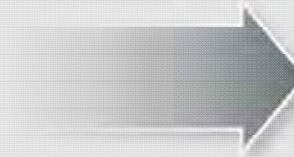
SCENARIO

TEST CONDUCTED

RESULT



Remove all 37 copied packages



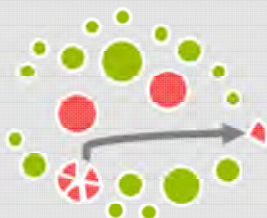
FAILED



Remove individual copied packages



FAILED

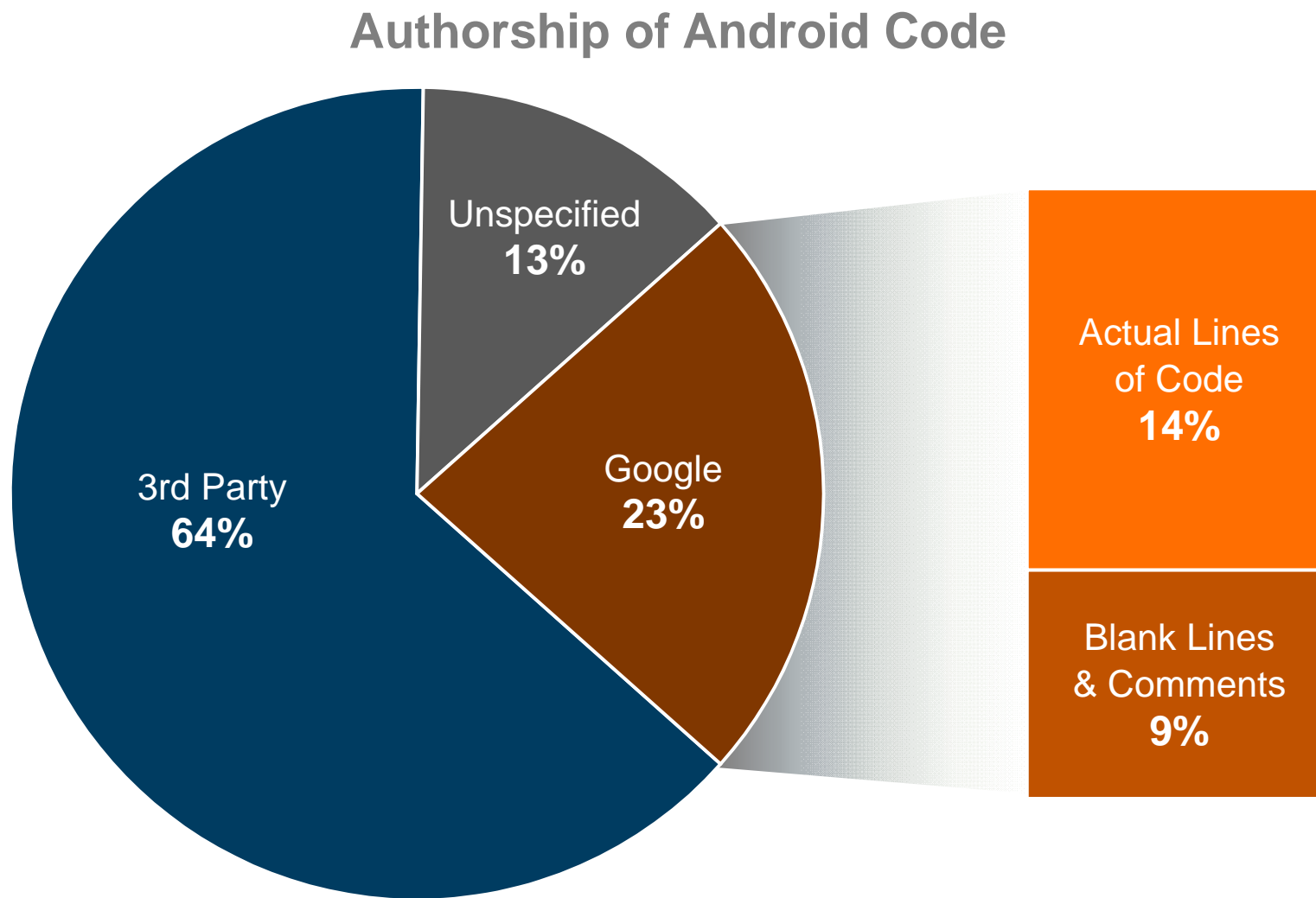


Remove copied lines of declaring code



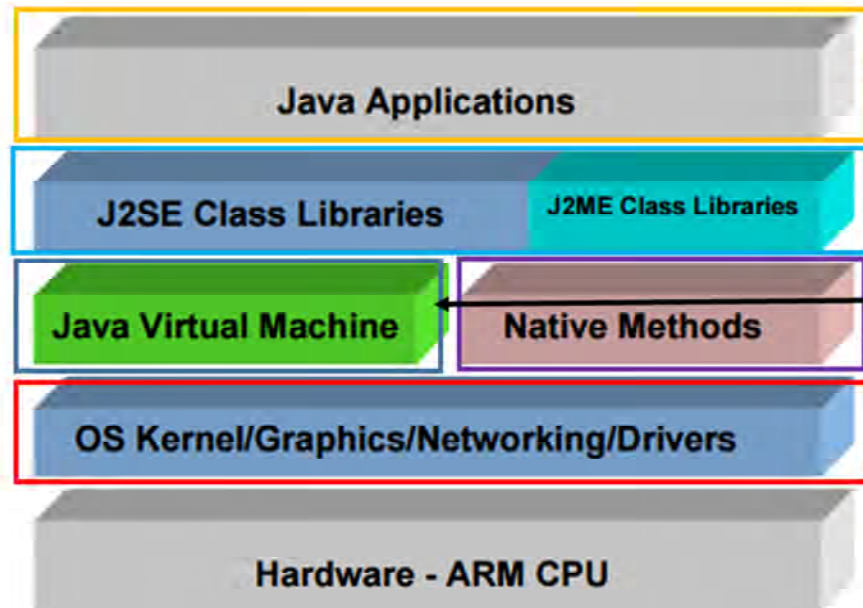
FAILED

Google Wrote Only 23% of Android Donut

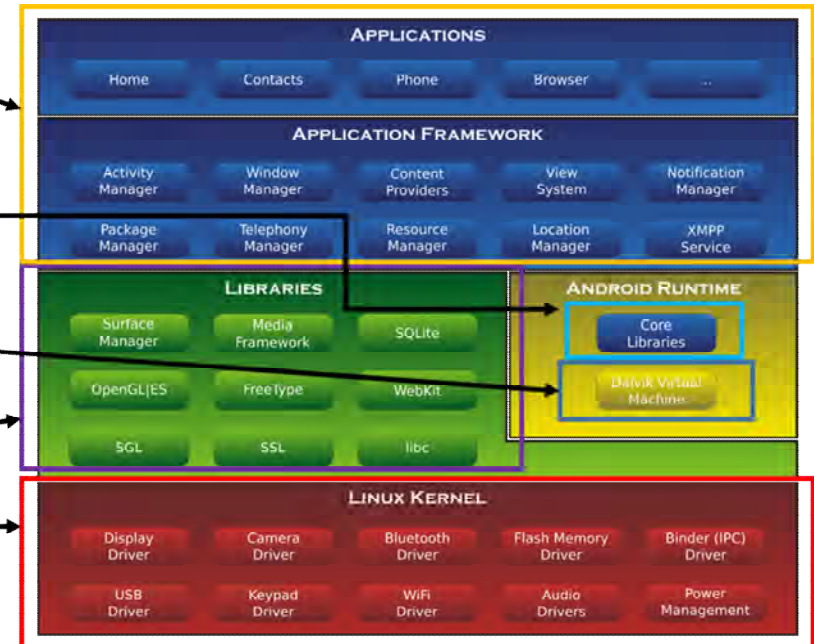


Sun's SavaJe OS and Android OS Have Similar Architecture

SavaJe OS Architecture



Android Architecture



Android Technology Has Java-Based Predecessors

Java used in mobile platforms many years before Android

T-Mobile Sidekick



HTC Dream



Operating System

DangerOS

Android

Uses Java?



Full keyboard, 'slider' form factor?



Runs apps?



Licensed Java?

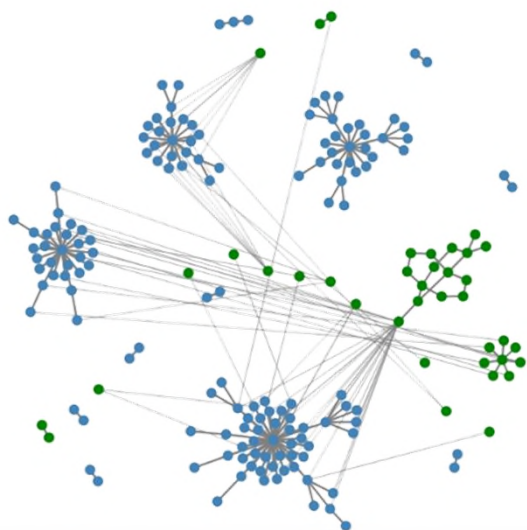


The Purpose of the Copied APIs Is Not Altered

java.security – Package Use Is Unchanged in Android vs. Java SE



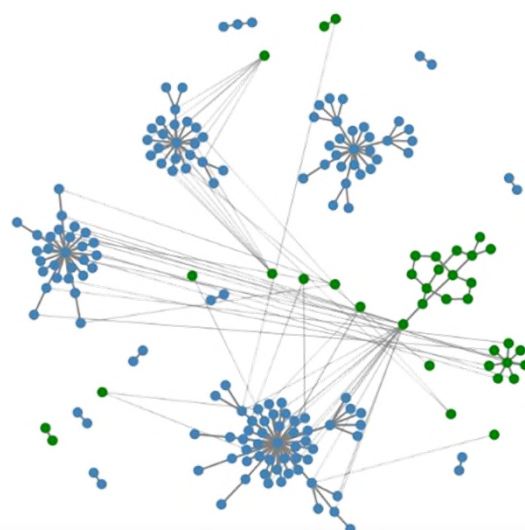
Java SE 5



“The java.security package, its subpackages...and its extensions... provide **classes and functionality related to security**, as the names suggest.”

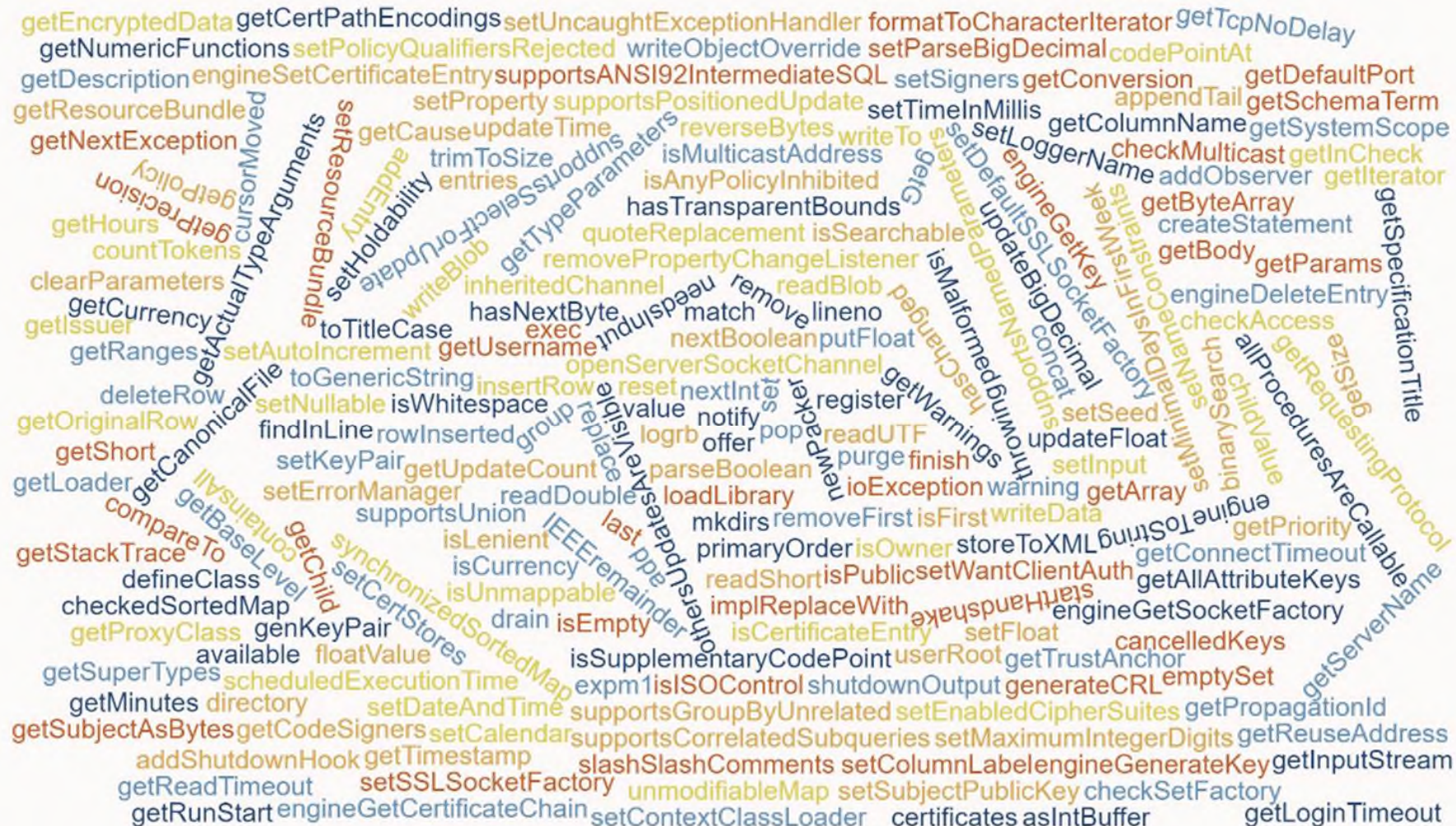


Android Gingerbread



“Provides **classes and interfaces for the security framework**. This includes classes that implement...access control security architecture”

There Is Diversity and Creativity in Class and Method Names



Design of Java APIs is Unique and Creative

EXAMPLE: Four options for Collections Libraries

Collections Libraries: Create/manipulate data (e.g. lists)

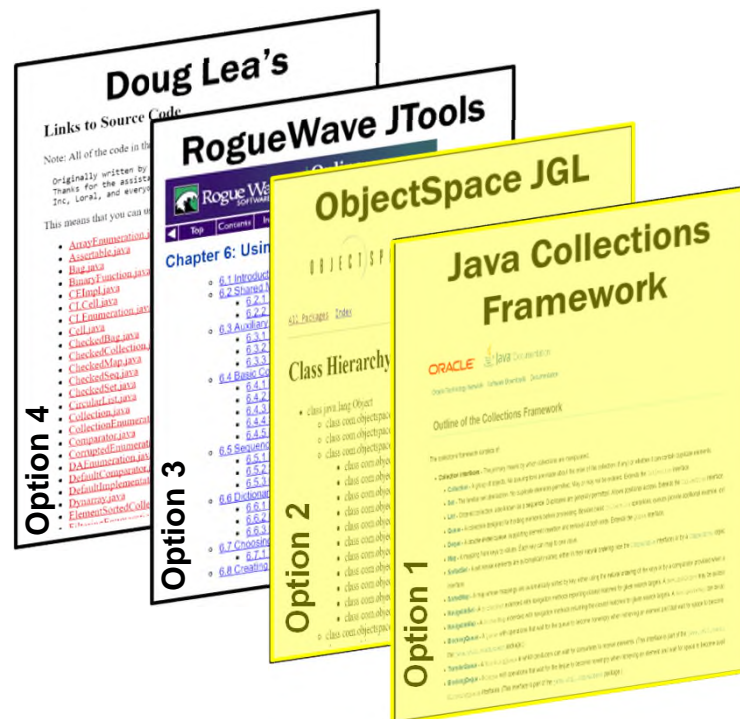
Java Collections Framework
(1998) – Single Package

java.util

**SIMILAR
COLLECTIONS
CAPABILITIES**

ObjectSpace JGL
(1996) – 6 Packages

com.objectspace.jgl
com.objectspace.jgl.adapters
com.objectspace.jgl.algorithms
com.objectspace.jgl.functions
com.objectspace.jgl.predicates
com.objectspace.jgl.util



Java and Android Applications Are Not Interoperable

**Java apps
do not run
on Android**



**Android apps
do not run
on Java**



Android's Copying Breaks Compatibility with Java SE

JCK Signature Test: Industry Benchmark Test of Java SE Compatibility

Android Lollipop makes **incompatible modifications** (to copied 37 APIs)

Package Name	Class		Error Type		Other		Total
	Added	Missing	Method	Added	Missing	Added	
java.awt.font	1	16		0	1	7	25
java.beans	0	32		0	0	0	32
java.io							110
java.lang							359
java.lang.annotation							10
java.lang.reflect							33
java.net	5	0		57	16	1	89
java.nio	1	0		22	1	0	24
java.nio.channels	7	0		37	0	0	54
java.nio.channels.spi	4	0		0	0	0	4
java.nio.charset	1	0		12	0	0	13
java.security	7	1		62	16	2	88
java.security.acl							
java.security.cert							
java.security.spec							
java.sql							
java.text							

Missing Class

Java SE Android

`public final class` **X**

`GlyphMetrics{`

Added Method

Java SE Android

`public static Policy`
`getInstance`
`(String type, Policy`
`Parameters params) throws`
`NoSuchAlgorithmException{`

X **✓**

FAILED

JCK for Java SE 5 against Android Lollipop

Summary of Opinions

- The copying is **not transformative**
 - Android uses the declaring code and design of the 37 Java APIs in the same way as the Java platform
 - Java-based mobile platforms already existed before Android
- The copied declaring code and design **are creative**
 - The copied declaring code and design reflect creative choices and could have been achieved very differently
- The copied packages are **very substantial**
 - The heart of the Java SE5 platform is found in Android
- The copying creates an **incompatible fork** of the Java Platform